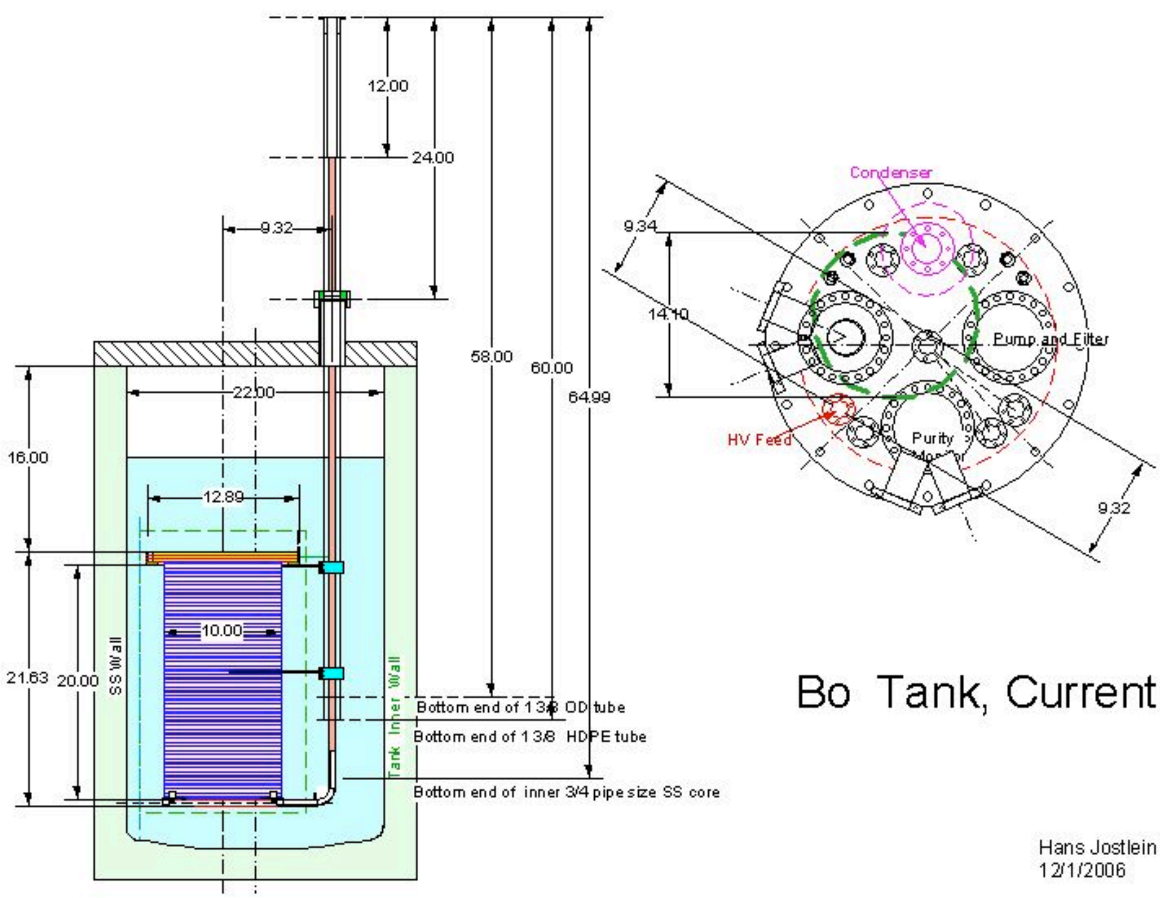
Bo TPC Readout Configuration

Hans Jostlein,   Dec. 7, 2006

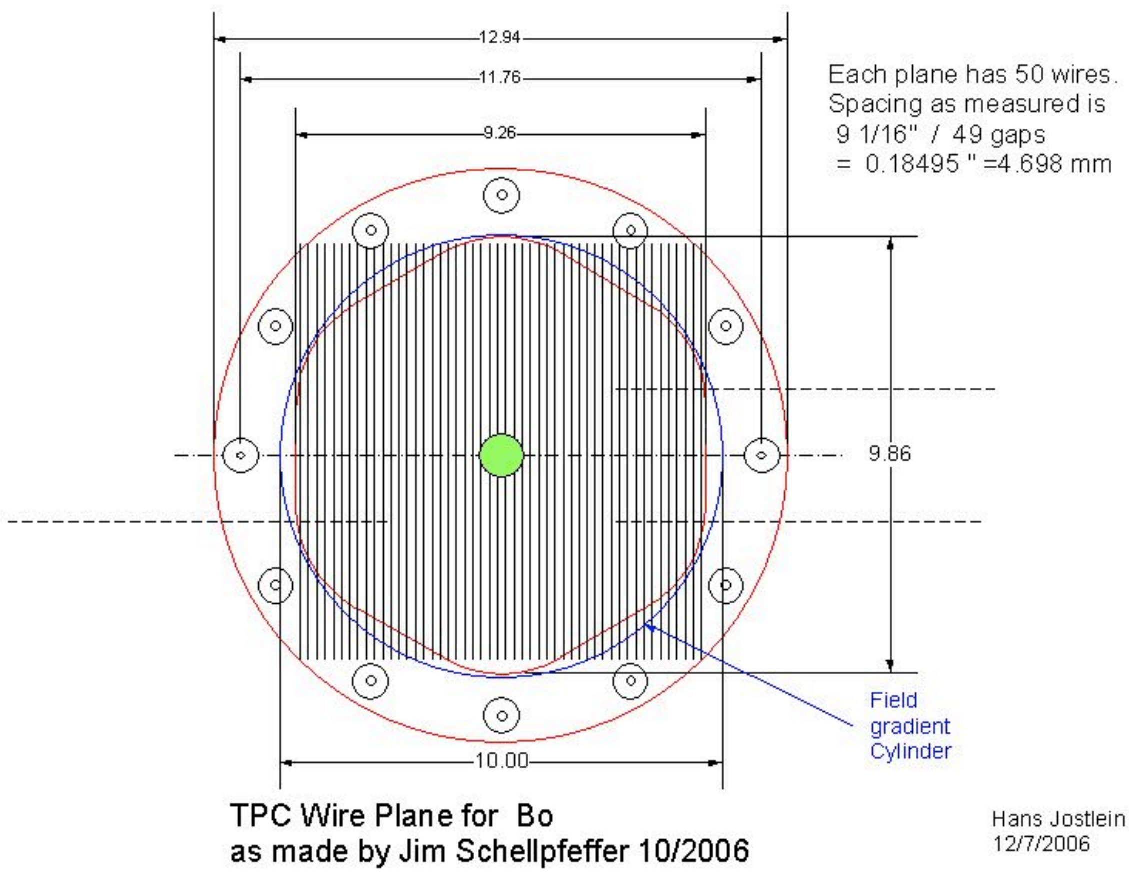
Introduction

The TPC being built for the “Bo” cryostat is sized to fill the available space in the cryostat, as shown below:



The space accommodates a TPC chamber outline (not counting the HV lead of about 13" diameter, with a 1/2 inch clear space around it.

We have already CNC cut files to make Chamber frames with an OD of 14 inches, designed for a previous cryostat (and used at Yale now).
There were 50 wires with a spacing of 0.200 inches (=5.08 mm). The odd spacing allowed a ¼-20 threaded rod to be used to place the wires.
Rather than re-programming, we scaled the frame to an OD of 12.95 inches, resulting in a wire spacing of 4.698 mm, still with 50 wires per plane:



Meanwhile, a project was started by Dan Edmunds to provide electronics for yet another TPC, this one designed small enough to fit in the Luke cryostat along with the materials test station , Argon filter and purity monitor. This chamber was designed to have 32 wires per plane.
Hence the electronics system under construction is designed for 32 wires per plane, or a total of 96 wires.

Question:

How do we connect 150 wires to 96 Electronics Channels ?

Choices:

- make chamber frames with 32 wires per plane
    - Smaller chamber (loose data) or
    - Larger wire spacing (requires larger plane spacing, hence higher interplane voltages)
- Make more electronics channels
    - Costly
    - Delays completion
    - Does not enhance experience to be gained
    - Not much gain in performance
- Connect only 96 wires to electronics Channels
    - OK, but with reduced tracking volume
- Wire up only 2 of the 3 planes
    - We loose capability for 3-plane reconstruction studies
- Connect multiple wires to channels
    - Pairs of wires: abnormal pulse shapes, poor reconstruction
    - Distant wires ganged on each channel:

→ **This proposal !**

Configuration Proposal

We propose to connect, in each of the three planes, the first 32 wires to the 32 channels that are allotted to each plane. Then we connect wire # 33 to electronic channel 1 (in addition to wire # 1). This is followed by wire # 34 being added ti channel 2, and so on until wire # 50 is added to channel # 18.

This leaves the middle 14 wires going to just one channel each (channels 19 through 32), while the leading and trailing 18 wires go in pairs to channels 1 through 18.

This sounds complicated, but is easy to implement>

If we use flat cable with ,mass termination connectors, we can place the connector somewhere in the middle of a cable, and solder the wires at both ends to different wire contact lands.

Reconstruction Problem ?

Let's call the middle 14 wires the "unique" wires, and the leading and trailing 18 wires the "degenerate" wires, for obvious reasons.

Track reconstruction can originate, in each plane where the track is registered by at least some of the "unique" wires. It will be very clear, for most tracks, which half of the "degenerate" wires saw the signal. This assumption will allow reconstruction of the track using all signals with no ambiguity.

If one plane does not favor a clear choice, recourse to the other 2 planes should resolve the ambiguity in essentially all cases.

A 96 channel system is going together.

Dan Edmunds,   Dec. 1, 2006

Its basic parts are:

- **6 cards called PMB-16** which hold 16 channels each of preamplifiers, postamp-filters, and the bias Voltage networks. These mount in a copper and then steel box right on the cryostat. I have one of these cards fully assembled and running now and 9 more are at the assembly house now.

- **A low noise power supply** to run the 6 PMB-16 cards. This is all put together and running now with an isolation transformer and DC power cable to connect it to the PMB-16 box on the cryostat.

- **3 cards called ADF-2** which hold 32 channels each of ADC and circular buffer and VME readout. I have these cards now. They sit in a Wiener VME-64x crate which I have now. The firmware that I'm currently using in the ADF-2 cards implements a circular buffer of 2048 steps with 198 nsec between steps so that it holds a 405 usec long record of data. These ADF-2 cards and the Wiener VME crate are spares borrowed from D-Zero. The firmware for the ADF-2 cards is specific to the LAr TPC application.

- **1 card called SCLD** which holds an SCL Receiver mezzanine card and a small hand made "Trigger Signal Receiver" mezzanine card. This SCLD card controls the timing of the ADF-2 cards, i.e. both the timing of their ADC samples and the timing of when the writing into their circular buffers stops. This SCLD card will receive the "Trigger Signal" from the "Muon Counter NIM Electronics", delay it for a digitally controlled period between 0 and 500 usec in steps of 2 usec, and then tell the ADF-2 cards to stop writing into their circular buffers.

This all exists and is working now.

The SCLD card mounts in the VME crate along with the ADF-2 cards.
The basic SCLD card is a spare borrowed from D-Zero and this rest of this setup is hand made and uses firmware specific for the Tiny LAr TPC DAQ application.

- **A "Bit-3" interface card** set to connect a PC computer to the VME crate. This consists of a card that plugs into slot #1 of the VME crate and an optical cable that ties it to another card that plugs into a PCI slot in the PC computer. MSU has one of these that Tiny LAr TPC DAQ may use for now. This "Bit-3" setup is in common use at Fermi. Bit-3 was purchased by SBC and now just recently purchased by GE.

- **A PC computer** to run the Tiny LAr TPC DAQ, either billy boy or Linux operating system. MSU can provide a computer for a while to get things running. I know nothing about what Fermi will want to do to this computer before allowing onto their network. Some one at Fermi will need to take care of setting it up to make Fermi happy with it. The Tiny LAr TPC DAQ and event data display software can run from a non privileged account under either operating system (but I would like to know which operating system you would like me to deliver).

- Software for the DAQ computer:

1. **Collect Data** i.e. "taking a run" All of the bricks for this DAQ software exist. Right now I have to go through a number of steps by hand to take and record each event. This will be scripted with Python. I hope to have this under way by the end of this week.

2. **Event Data Display.** The display I have put together has time on the horizontal axis and the DAQ channels are offset from each other along the vertical axis. My assumption was that we would have 32 channels in each of the 3 planes in the detector and that I would make a separate 32 channel plot for each plane. You can see two examples of 16 channel plots at:

   www.pa.msu.edu/~edmunds/LArTPC/PMB_16/event_1_ch_0_15.pdf
   event_2_ch_0_15.pdf

   This is NOT an event display - it is just something to let one see what the raw data looks like. There are only 16 channels because I only have one assembled PMB-16 card at this time. In each of these plots, just one channel has a real size pulser signal in it and the other 15 channels have no input.

   Right now I have to go through a number of steps by hand to make each plot. This will be scripted with Python, i.e. you will be able to browse for the event that you want to plot then click it and up comes 3 plots of 32 channels each. I have not started this script yet but we have similar scripts for making similar plots on other systems.

All of this software stuff is leveraged from other things we have done and 99% of the work is done by Philippe Laurens at MSU.

- I have put some not very good pictures of the DAQ system hardware components on the web at:

  www.pa.msu.edu/~edmunds/LArTPC/

  pict_adf_2_card.jpg
  pict_adf_2_card_and_vme_crate.jpg
  pict_pmb_16_card.jpg
  pict_preamp_power_supply.jpg
  pict_scld_card.jpg

Draft of what the Tiny LAr TPC DAQ system will look like to the user

Please comment or make any suggestions that you want. The intent of this DAQ system design is to quickly make something very simple for a small number of channels running at a low rate. That is, get something running so we can learn what we want/need. The human interface is:

- Walk up to it and click on **"I want to make a new Run"**

- It will ask you, "**Which channels** do you want to readout in this run ?". Including/excluding channels is controlled at the ADF-2 card level, i.e. at the 32 channel level which I assume will be the same as at the detector plane level. See the proposed channel map below.

- **It will ask you, "Do you want to Auto-Pause or PreScale ?".**

  Auto-Pause means that after each trigger, it reads out that trigger's worth of data and then Pauses until you Resume the run. We find this collecting one event at a time mode to be very useful when debugging.

  Prescale means that for each trigger it reads out that trigger's worth of data and then "sleeps" for 1 to N seconds before waking up and being sensitive to another trigger. This is useful in a high trigger rate environment when you just need to collect a sample of the data to study noise or something.

--> I have no idea what rate we expect from the
--> Muon Counter NIM Trigger or what fraction of these
--> triggers will be from real muons going through
--> the sensitive volume of the Tiny LAr TPC ??

- It then creates a new directory to hold all of the events from this run. The filename of this directory includes the Run Number and the date and time when this directory was created. Each event in this run will be a separate file in this directory. The filename of each event data file contains the Run Number, the Event Number, and the time when that event was recorded.

- **The run now starts** and with each accepted trigger signal the appropriate set of ADF-2 cards will be readout and the data written into an event data file. The proposed data format is described below.

- If you have selected Auto-Pause for this run then you must manually click Resume after each event to make the system sensitive to a new trigger.

- In any case you can manually click to Pause and then later click to Resume the run at any time that you want to.

- **When you are finished** you click End to end this run.

- There is a status box that tells you the current state of the daq system (e.g. is it in a run or not is it paused or running if it is in a run then it displays the run number and the number of events that have been recorded).

That is all that we propose to have for the human interface to this simple DAQ system.

The GUI to browse for and then select an event that you want to make plots for is a separate piece of software from the DAQ system. It can be run at the same time as the DAQ system but it is a separate chunk of software.

Proposed Event File Data Format

I would like to keep this as simple as possible. I don't think that we need any headers or trailers or checksums or any of the "self described data" type stuff that one uses in big systems.

The following are the details of the Proposed Event File Data Format:

- All event data files from a given Run are in a separate sub-directory. The filename of that sub-directory includes the Run Number and the time when that run was started.

- There is a separate event data file for each event

- For each event data file its filename includes the Run Number, the Event Number, and the time that the event was recorded.

- There are no header or trailer records at the top or bottom of the event data file.

- The data from each channel that is being readout begins with a one line comment in ascii giving the Channel Number that the following data came from.

- Readout data is ascii, 4 characters per line, begins in column number 1, and is right justified. Each line is the data from one ADC sample. The value is in the range 0:1023 and is unsigned decimal data.

- The pedestals for all channels in the event data files is set to some defined fixed convenient value for the ADF-2 hardware, e.g. 400.

- When writing each data file the circular buffer ADC data is layed out in continuous time order, the oldest ADC sample is the first sample in the data for a given channel and the newest ADC data is the last sample in the data for a given channel.

- There are 2048 ADC samples in the data from each channel.

  --> Note if the Tiny LAr TPC has a drift time through its
  --> sensitive detector region that is significantly shorter
  --> than 400 usec then we could record fewer than 2048 samples
  --> from each channel in the event data files.

Proposed Channel Numbering:

- Channels 0:31 First Induction Plane ADF-2 card in Slot #10
- Channels 32:63 Second Induction Plane ADF-2 card in Slot #11
- Channels 64:95 Collection Plane ADF-2 card in Slot #12

The Bit-3 card is in slot #1
The SCLD card is in slot #19
The External Trigger Input is in slot #21

QUESTIONS:

- Length of Drift Time in the Tiny LAr TPC Detector ?

  Is recording and ADC sample every 198 nsec OK ?
  How long of a time record do you want
  recorded in the event data files ?

- Trigger Rate and Quality of the Triggers from the Muon Counters ?

  I don't know what is expected here. Clearly if only a small percentage of the triggers will have a real muon going through the sensitive volume of the detector then we will need to look through a lot of data to find tracks.

- Cryostat to DAQ Electronics Rack Distance ?

  The Pleated Foil Cables that I have to connect the analog signals from the PMB-16 cards in the box on the cryostat to the ADF-2 cards in the VME crate in the DAQ electronics rack are only 3m long. Is that long enough to have the DAQ Electronics Rack at a convenient distance from the cryostat or do you want me to purchase longer PFC cables ?

- Cryostat Feed Through and Input Cables

Is there anything that I need to be doing about this ?

I received the note from Hans and Walter that you have found a Halogen free twisted pair cable to use inside the cryostat. That's great.

If we know what connector is on the outside side of the feed through and if we know the feed through to PMB-16 box distance, then I will get started making the cables to connect the feed through to the PMB-16 inputs.

Other Comments:

- I will put together a small battery operated 16 channel pulser with the output from each channel delayed from the channel before it by a couple of usec. The intent is to be able to test the full signal chain before closing the cryostat and to be able to record some test event data files that have sections that look a little like a real track. Doug, this is probably the first chance to give you data that looks a little bit like a track. I assume that I should make this pulser so that it can plug into either the inside side of the feed through or directly into the PMB-16 cards ?

- I have run the PMB-16 card with 500 Volts of wire Bias Voltage. This looks OK. The DC blocking capacitors and Bias Voltage filter capacitors that I purchased for the PMB-16 cards look OK, i.e. no extra noise when these capacitors are holding off 500 Volts. They make a lot of random excess noise for 10 or 20 seconds after the Bias Voltage is turned on or a big change is made in its value. This was tested with a good quality lab type power supply providing the 0 to 500 Volts. Do we know what supplies will be used for Bias Voltage at PAB ?

Dan